REMARKS

Claims 1-29 are pending in the above identified application. The Examiner has rejected claims 1-29. Applicants herein traverse these rejections.[1] Independent claims 1, 18, and 25 have been amended to clarify that the "plurality of classes" refers to application code as opposed to system resources.

Rebuttal of Examiner's Response to Arguments

The Examiner has indicated that "Applicant's arguments filed 05/08/2006 have been fully considered but they are not persuasive." As the Examiner characterizes Applicant's position, "Reference V only teaches permissions to access resources and not permissions to access a "first trusted class", and therefore Reference V fails to teach controlling access to a first trusted class by untrusted class or a second trusted class based upon privilege information associated with the first trusted class." (Office Action, p. 2) As further stated by the Examiner

> Examiner would point out that the term 'class' in object oriented programming refers to classes, objects, threads and processes [see also, specification page 6, paragraph 022]. In this case, reference V teaches Domain-Based access Control, wherein classes, objects and threads belong to a certain domain [see section 2.3, 1st paragraph]. Reference V further teaches interaction between multiple domains, for example, application domain and system domain (see page 5, section 2.4, paragraphs 3-5, where it states "*...where a system domain invokes a method from an application domain, such as when an AWT system code calls an applet's paint method to display the applet, it is again crucial that at any time the effective access rights are the same as current rights enabled in the application domain.*" Examiner would point out that, a method/function in object-oriented programming is part of a class. In this case invoking/calling a method/function from one domain to another domain by controlling access rights, means, access to the class itself by another class in a different domain.

[1] Characterizations of both the claims of the present application and the teachings of various prior art are made throughout the Office Action. Applicants do not automatically agree or acquiesce in any of these characterizations, even if they are not specifically addressed in this response.

> Examiner asserts the art on record teaches the claim limitations and therefore respectfully maintains the rejection.

(Office Action, p. 2). In order to clarify the invention, Applicants propose to amend independent claims 1, 18, and 25 to clarify that the "classes" are classes of "application code" and do not refer to system code. Claim 1 has been amended to recite "separating a plurality of classes of application code into at least a first trusted class and an untrusted class," claim 18 has been amended to recite "a first memory space for storing an untrusted class of application code; a second memory space for storing a first trusted class of application code," and claim 25 has been amended to recite "separating a plurality of classes of application code into at least a first trusted class and an untrusted class." Therefore, the independent claims teach controlling access between application codes, and do not address access achieved by system resources.

As pointed out in Applicants' remarks of 05/08/2006, and as discussed by the Examiner, Reference V controls access to resources and not access between application code. As the Examiner points out above, Reference V teaches interaction between different domains, which can include system resources interactions with applications programs as is described in Reference V, paragraph 2.4, paragraph 3. However, the permissions taught in reference V are permissions to access system resources and not permissions to access other application code. As is further stated in that paragraph, Reference V discloses

> A less "powerful" domain cannot gain additional permissions as a result of calling a more powerful domain; whereas a more powerful domain must lose its power when calling a less powerful domain. This principle of lease privilege is applied to a thread that transverses multiple protection domains.

(Reference V, sec.. 2.4, 5th par.). Further, Reference V teaches the permissions granted by code in one domain that calls code in another domain. It does not teach or address granting permission for code in one domain to actually call code in another domain.

Therefore, Reference V does not teach "separating a plurality of classes of application code into at least a first trusted class and an untrusted class," as recited in claim 1, "a first memory space for storing an untrusted class of application code; a second memory space for storing a first trusted class of application code," as recited in claim 18, or "separating a plurality of classes of application code into at least a first trusted class and an untrusted class," as recited in claim 25.

Claim Rejections Under 35 U.S.C. § 103

Claims 1-6, 10-21, and 25-29

The Examiner has rejected claims 1-6, 10-21, and 25-29 under 35 U.S.C. 103(a) as being anticipated by WO 99/30217 ("Gong") in view of Gong et al. "Going Beyond the Sandbox: An Overview of the New Security Architecture in the Java Development Kit 1.2" (hereinafter Reference V).

As the Examiner points out, Gong does not teach controlling access to the first trusted class by the untrusted class or a second class based upon the privilege information associated with the first trusted class. (*See*, Gong, page 3 and Gong, page 4). The Examiner then relies on Reference V to cure the defects in the teachings of Gong. As pointed out above, however, Reference V does not cure this defect in the teachings of Gong because Reference V teaches access to resources afforded code in one domain that calls code in another domain and does not address access to a "trusted class of application code" by an "untrusted class" of application code as recited in the claims.

Therefore, the combination of Gong with Reference V does not teach "separating a plurality of classes of application code into at least a first trusted class and an untrusted class; . . . and controlling access to the first trusted class by the untrusted class or a second trusted class

based upon the privilege information associated with the first trusted class," as is recited in claim 1; "a first memory space for storing an untrusted class of application code; a second memory space for storing a first trusted class of application code; . . . and a controller for controlling access to the first trusted class during a trusted class operation," as is recited in claim 18, or "separating a plurality of classes of application code into at least a first trusted class and an untrusted class; . . . and controlling access to the first trusted class by the untrusted class or a second trusted class based upon the privilege information associated with the first trusted class," as is recited in claim 25. Independent claims 1, 18, and 25 are thus allowable over the combination of Gong with Reference V.

Claims 2-6 and 10-17 depend from claim 1 and are allowable over the combination of Gong with Reference V for at least the same reasons as is claim 1. Claims 19-21 depend from claim 18 and are allowable over the combination of Gong with Reference V for at least the same reasons as is claim 18. claims 24-29 depend from claim 25 and are allowable over the combination of Gone with Reference V for at least the same reasons as is claim 25.

Claims 7-9 and 22-24

The Examiner has rejected claims 7-9 and 22-24 under 35 U.S.C. § 103 over the combination of Gong, Reference V, and "Extending Java for Package Board Access Control," IEEE 2000, pp. 67-76 (hereinafter Papa et al.). As discussed above, claims 1 and 18 are allowable over the combination of Gong and Reference V. Papa et al. does not cure the defects in this combination. Therefore, claims 1 and 18 are allowable over the combination of Gong, Reference V, and Papa et al. Claims 7-9 depend from claim 1 and are allowable over the combination of Gong, Reference V, and Papa et al. for at least the same reasons as is claim 1.

Claims 22-24 depend from claim 18 and are allowable over the combination of Gong, Reference V, and Papa et al. for at least the same reasons as is claim 18.

Conclusion

In view of the foregoing amendments and remarks, Applicant respectfully requests reconsideration and reexamination of this application and the timely allowance of the pending claims.

Applicant respectfully requests that this Amendment under 37 C.F.R. § 1.116 be entered by the Examiner, placing claims 1-29 in condition for allowance. Applicants submit that the proposed amendments of claims 1-29 do not raise new issues or necessitate the undertaking of any additional search of the art by the Examiner, since all of the elements and their relationships claimed were either earlier claimed or inherent in the claims as examined. Therefore, this Amendment should allow for immediate action by the Examiner.

Furthermore, Applicants respectfully point out that the final action by the Examiner presented some new arguments as to the application of the art against Applicant's invention. It is respectfully submitted that the entering of the Amendment would allow the Applicants to reply to the final rejections and place the application in condition for allowance.

Finally, applicants submit that the entry of the amendment would place the application in better form for appeal, should the Examiner dispute the patentability of the pending claims.

In view of the foregoing remarks, Applicants submit that this claimed invention, as amended, is neither anticipated nor rendered obvious in view of the prior art references cited against this application. Applicants therefore request the entry of this Amendment, the Examiner's reconsideration and reexamination of the application, and the timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: September 25, 2006

By: _____
Gary J. Edwards
Reg. No. 41,008

EXPRESS MAIL LABEL NO.
EV 901562695 US